Landsat 7 Processing System (LPS) Integration and Test Plan

March 18, 1996

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

Landsat 7 Processing System (LPS) Integration and Test Plan

March 18, 1996

Prepared by:

Ernest Staples 4/25/96
Ernest Staples Date
Systems Test Engineer
Landsat 7 Processing System
SEAS/Computer Sciences Corporation

Reviewed by:

J. Williams 25 APR 96
John Williams Date
Section Manager
Baseline Engineering and System
Test Department
SEAS/Computer Sciences Corporation

Concurred by:

Tanweer Aslam 4/24/96
Tanweer Aslam Date
System Development Manager
Landsat 7 Processing System
SEAS/Computer Sciences Corporation

Concurred by:

Dana Ferguson 4/26/96
Dana Ferguson Date
Deputy Systems Engineer
Landsat 7 Processing System
Code 514
Goddard Space Flight Center

Concurred by:

Robert Schweiss April 26, 96
Robert Schweiss Date
Systems Engineering Manager
Landsat 7 Processing System
Code 514
Goddard Space Flight Center

Approved by:

Joy Henegar 4/29/96
Joy Henegar Date
Project Manager
Landsat 7 Processing System
Code 514
Goddard Space Flight Center

Abstract

This document provides a comprehensive plan for the subsystem integration and system-level testing of the Landsat 7 Processing System (LPS). This plan documents the process from test preparation through test status summary reporting, and includes test methodology, functional responsibilities, test resource requirements and constraints, and test set information.

Table of Contents

Section 1 - Introduction

Section 2 -- Test Activities

Section 3 -- Subsystem Integration Testing

Section 4 -- System Testing

Section 5-- Configuration and Data Management

Section 6 -- Test Specifications

Section 7 -- Test Resource Requirements

ACRONYMS

LPS System Test Verification Matrix

Section 1 --Introduction

In support of the National Aeronautics and Space Administration (NASA) contract NAS5-31500, this test plan describes how Computer Sciences Corporation (CSC) performs subsystem integration of the Landsat 7 Processing System and executes system-level tests to verify compliance with the system requirements allocated to the LPS.

The integration and test plan complies with the policies, methodologies, and directives of the Systems, Engineering, and Analysis Support (SEAS) program, in accordance with the SEAS System Development Methodology (SSDM).

1.1 Purpose and Scope

This document provides a comprehensive plan describing how the Mission Integration Technology Group (MITG) integrates LPS subsystems, and performs tests to qualify the system for release to the Assistant Technical Representative (ATR).

This plan covers the verification of LPS system requirements, internal interfaces at the subsystem level, and external interfaces with other elements of the Landsat Ground System. The overall Landsat 7 test hierarchy and responsibilities, including the Landsat 7 Ground Station (LGS), and the Land Processes Distributed Active Archive Center (LP DAAC), can be found in the Landsat 7 Ground System Integration and Test Plan.

1.2 LPS Overview

The LPS is one of five main elements of the Landsat 7 ground system. The primary functions of the LPS include capturing raw wideband data transferred in realtime from the LGS; writing the data to removable media for short term storage; performing Level 0R processing; and making the output files available for transfer to the LP DAAC.

Data are transmitted during the time that the Landsat 7 satellite is in view of the LGS; these contacts will occur for two or three orbits in a row, twice a day. The LPS will be operational 24 hours a day, 7 days a week, to receive and process data from these contacts.

1.3 Applicable Documents

These documents were referenced in developing the LPS Integration and Test Plan.

1. CSC, Mission Integration Technology Group (MITG) Configuration Management Plan, Review, August 8, 1995.

2. CSC, SEAS System Development Methodology (SSDM), Copyright July, 1989.

3. Martin Marietta Astro Space (MMAS), Landsat 7 System Data Format Control Book (DFCB), Volume 4 - Wideband Data, Revision B 23007702-IVB, May 15, 1995

4. NASA GSFC, Interface Control Document (ICD) between the EOSDIS Core System (ECS) and the Landsat 7 System, Working Draft, 209-CD-013-001, June 1995

5. NASA GSFC, Interface Control Document (ICD) between the Landsat 7 Ground Station (LGS) and the Landsat 7 Processing System (LPS), Signature Copy, May 26, 1995

6. NASA GSFC, Memorandum of Understanding (MOU) between the Landsat 7 Processing System and the Mission Operations Center (MOC), May 1995

7. NASA GSFC/MO&DSD, Landsat 7 Ground System Integration and Test Plan, Draft Copy, 510-2ITP/0395, October, 1995.

8. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) Build Implementation Plan, Draft, 514-84BIP/0195, August 2, 1995.

9. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) Functional and Performance Specification, Revision 1, 560-8FPS/0194, July 14, 1995.

10. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) Interface Definitions Document, Draft, 560-1IDD/0195, April, 1995.

11. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) Operations Concept, Revision 1, 560-30CD/0194, July 14, 1995.

12. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) Software Requirements Specification, 560-8SWR/0194, April 28, 1995.

13. NASA GSFC/MO&DSD, Landsat 7 Processing System (LPS) System Design Specification, 560-8SDS/0194, May 26, 1995.

14. NASA GSFC/MO&DSD, Performance Verification Plan for the Landsat 7 Ground System, Revision 1, Review, November, 1995.

1.4 LPS Integration and Test Strategy

In order to achieve "faster, better and cheaper" testing of the Landsat 7 Processing System, project personnel will be "reused" through various stages of the LPS lifecycle. System engineering staff will transition to development, to integration test, and/or to system test positions. This approach will ensure coordination among the groups, the adoption of a "big picture" view of the system, and that project knowledge is not lost or diminished through the development lifecycle of LPS.

Reuse is a major focus for testing LPS. Test tools and test data are defined to support unit testing, as well as integration and system testing. Operational scenarios and string testing techniques are used to provide tests that mimic realistic occurrences of a day in the life of the LPS.

Prior to the subsystem integration planned in this document, the LPS software will be unit tested during development; the tested units will be integrated into modules and the modules into subsystems by development personnel working as an integration test team. When the software has been integrated that far, the subsystems will be turned over to subsystem integration teat team as described in section 7.4.1 of this plan. Subsystem integration testing verifies the internal interfaces between system components. In addition to testing the interfaces between individual subsystems, string testing will be performed to flow data from end to end of the software in a given build. Problems identified during subsystem integration will be documented as Internal Configuration Change Requests (ICCRs) and corrected as time permits and the LPS team deems appropriate. An informal summary of the results of subsystem integration testing will be documented at the end of that test phase.

The purpose of system testing is to verify the requirements allocated to the LPS and documented in the Landsat 7 Processing System (LPS) Functional and Performance Specification (F&PS), to identify system errors, and to ensure that the system is ready for release to the ATR.

System integration tests are defined to exercise the LPS with respect to performance requirements and operational procedures. Test scenarios that mimic the operational system include verification of the electronic interfaces between the LPS and other elements of the Landsat 7 ground system. Implementation and test schedules of the LGS and the LP DAAC may impact this portion of LPS testing, however, tools to simulate the other ground system elements will be used, when required, to verify the LPS software.

Section 2-- Test Activities

This section presents an overview of testing and describes the typical activities performed by the test groups throughout the project lifecycle.

2.1 Test preparation

The test preparation phase consists of test plan development, test tool preparation, test data generation, training of test personnel, and test environment preparation. At the end of the initial test planning phase, and prior to system testing of each build, preparation includes test procedure definition, the Test Procedures Walkthrough (TPW) and the System Test Readiness Review (STRR).

The test plan for LPS addresses internal interface testing on the subsystem level, system-level testing and requirements verification, and external interface testing with other elements of the Landsat ground system. This document is the basis for all related test planning activities, including planning for test tool and test data needs.

Because lead test personnel will have transitioned through system engineering and development phases on the LPS, details of LPS application software implementation will be "built in" to the test phases without the need for formal training sessions.

Informal demonstrations and/or user's guides should be provided to test personnel for

- GTSIM
- GTEDIT
- DESIM
- HDF tools
- any test tool developed for LPS (see Section 7)
- LPS Operator Interface
- LPS database

More formal training may be required for Commercial-Off-The-Shelf (COTS) products and system software associated with LPS, and should include

- UNIX
- ORACLE
- Requirements & Traceability Management (RTM) tool
- Polytron Version Control System (PVCS) tool
- Revision Control System (RCS) tool

After implementation of the first build, the software in development may be Build n, while the integration team is testing Build n-1 and Build n-2 is the system test baseline. In order to manage and ensure control of the software

and environment configurations, integration and system test environments are separate from each other, and from development environments. For test environments, the LPS system engineers set up and configure all hardware elements and system software. Any modifications, including changes from standard installation process and/or the parameters that are set during the installation, made to the hardware or system software are documented for configuration control. The appropriate test group, with support from the configuration management (CM) representative, controls the application software for the corresponding test environment. The subsystem integration group installs software as it becomes available for test, and rebuilds (i.e., exercises the full build instructions) before turnover to system test. Application software for the system test environment is installed as a full build (from no existing software) for delivery of full releases. For partial releases (e.g., emergency deliveries outside the scheduled deliveries), only those delivered application software components are built.

Testing follows the development of the software components for each build and is, therefore, dependent on the build process for the LPS. The Landsat 7 Processing System (LPS) Build Implementation Plan describes the contents and the schedule of, as well as the specific requirements allocated to each build. The test cases corresponding to those requirements are the scope of the system test for that build (including requirements from previous builds). The incremental build approach provides feedback on quality, errors, and design concerns as early as the integration of the first build and facilitates improvements in subsequent builds.

Prior to each test execution phase, test procedures are developed to define the step-by-step execution and expected results of testing for a given build. Test procedures are red-lined during test execution to note any obvious variances from the original planning. After the test execution phase, the test procedures are updated and gathered as an appendix to the test plan, and used as a basis for procedures for subsequent builds.

Delivery package preparation begins in the test preparation phase. The turnover package started by development is the draft delivery package. The package is augmented with information from integration test and passed in the turnover to system test. The delivery package for a release is finalized with all attachments at the end of the system test for that release. Details of the delivery package contents are documented in Landsat 7 Processing System (LPS) Delivery Procedure for the LPS.

At the end of system test preparation the test procedures walkthrough is held to review test execution steps and expected output, and the system test readiness review is held to assess whether or not the project is ready to transition to system test.

2.2 Test Conduct

When test preparation is complete, the test group moves into the test conduct phase for the current build. The primary activities of this phase are test execution and analysis of the software functions and/or output files. In addition, status and progress are reported, and change requests are generated. For a build that will be released to the customer, the delivery package is finalized during the system test conduct phase.

2.3 Test Management

Test management of the LPS includes preparation of weekly test status reports, daily productivity status reports, and formal STRRs for system testing. A summary of the test execution, problem reports, and major milestones is provided in the weekly status reports, along with the overall progress of testing. The weekly test status reports are provided to SEAS management during the entire test life cycle and incorporated into the weekly LPS status reports provided to Goddard Space Flight Center (GSFC). Productivity status information is provided during test conduct and contains totals of tests executed and ICCRs written. A summary of the major problems encountered and major milestones completed for the day is included with this report. The report is provided to the LPS test lead and is available for LPS project management and GSFC, if desired. Meetings on the test status are held periodically during the test execution phase, as needed.

2.4 Internal Configuration Control

Maintaining configuration control of LPS software, tools, data, and environments is necessary to minimize the effort and risks inherent in developing and testing changeable software. The Configuration Management section of the MITG Baseline Engineering and System Test Department and the applicable test group are each responsible for configuration control. These responsibilities are described in further detail in Section 5 of this plan.

2.5 Test Data

Development, integration, and system test personnel require data to test individual units, modules, subsystems, and the LPS system as a whole. The Generic Telemetry Data Simulator (GTSIM) will be used throughout the LPS lifecycle to generate raw wideband data in the format defined for the Enhamced Thematic Mapper Plus (ETM+) instrument of the Landsat 7

satellite. Section 7 of this document contains a more detailed discussion of test data requirements and sources.

2.6 Test Tools

LPS testing will verify low-level tasks in level 0R processing, and will depend heavily on data dump and analysis tools. Integration test will require drivers and stubs to stand in for software scheduled for later development. To eliminate dependence on the development schedules of other facilities, LPS will develop interface simulators. Specifics on test tools and tool configuration are documented in Sections 7 and 5, respectively.

Section 3 -- Subsystem Integration Testing

This section describes the preparation for and execution of subsystem integration testing for the LPS application software.

3.1 Objectives

The integration test effort documented in this plan is designed to verify software subsystem interfaces, including the data passed in those interfaces. The integration team also tests the LPS software to verify accuracy in mathematical algorithm implementations, to exercise error handling paths and verify path coverage in general, and to test functional performance.

3.2 Test Cases

Integration tests are designed using the Interface Definition Document (IDD) to map interactions between subsystems. A class of valid inputs is identified for each interface, and tests are defined to use a random or representative subset of those values. Invalid inputs and boundary conditions are identified, and one test case may be used to verify nominal, boundary and error conditions by changing input criteria.

An integration test case consists of the objective of the test, a description of the test case, prerequisites and dependencies, initialization and input requirements for the test, expected output and test evaluation criteria.

3.3 Test Execution

Based on the contents of a build, the integration team will identify the test cases to be executed. Prior to the first turnover, integration test will hold an informal walkthrough to present the tests planned for the current build. A test is considered unsuccessful when the results do not match the expected output; early review and discussion of the integration test plan by development personnel is essential to streamline test execution.

Installation of software subsystems and test execution is performed in the subsystem integration test environment, separated and under separate configuration control from the development environment. Test execution is scheduled based on identified test dependencies and the order in which subsystems are turned over to the integration team.

3.3.1 Internal Interfaces

Unit and module integration is completed by the development team prior to software turnover to subsystem integration testing team. The team tests the interfaces between subsystems, including interfaces between LPS subsystems and external elements of the Landsat 7 Ground System (NOTE: external interfaces will be simulated for subsystem integration).

Integration tests will verify the interfaces between and among subsystems (i.e., each interface between two subsystems, and all interfaces among a group of subsystems), until the implemented software has been fully integrated. For early builds, when implementation of a given subsystem is not complete, test tools will substitute for undelivered units.

3.3.2 String Testing

Software turnovers to subsystem integration are expected to be staggered according to the development schedule for each subsystem. The last turnover of an LPS subsystem must occur at least one week before turnover to system test. When the subsystems have been integrated, tests will be executed to verify the data and process flows from initial input to the LPS through final output from the LPS.

The Landsat 7 Ground System Integration and Test (I&T) will be supported by released or engineering versions of the LPS software (dependent on Ground System I&T schedules). For subsystem integration, external elements may be simulated to verify the LPS software at those interfaces and to flow data from end to end of the LPS.

3.3.3 Confidence Tests

At the end of the integration test phase, and in preparation for system test, the two test teams will execute confidence tests. Configured data sets are processed by the LPS software in the integration test environment. Behavior of the software and data results are recorded. After the LPS software is installed in the system test environment, the same configured data sets are processed in that environment. Results are compared to integration test results to obtain a quick confidence level in the installation process and software.

3.4 Analysis and Reporting

The level of analysis required to determine the status of an executed test will vary with the test. The expected results documented in each test case description range from observation of an event to analysis of data integrity for a file. Any test whose results fail to match the expected output is an unsuccessful test, and the integration team will refer such anomalous or unexpected test results to the software development team for further analysis.

An informal, daily report of integration test progress and unresolved issues will be prepared by the test team for project management. A more formal accounting of tests executed versus tests scheduled; successful and unsuccessful test numbers; number, priority and type of ICCRs written and resolved; and status of integration test issues will be prepared on a weekly basis.

3.5 Problem Reporting

Software problems encountered during the subsystem integration test phase will be documented as ICCRs, using the Iteractive CCR Automation System (ICAS) system. The ICCR process is simplified and controlled internally and the time to correct errors can be minimized without compromising configuration control in the integration environment. However, all the unresolved ICCRs from subsystem integration test will be placed under CRB control after software turnover to system test.

The number of procedural and set-up errors, or errors due to implementation changes, should be reduced by the test walkthrough. In order to eliminate unnecessary or incorrect error reporting even further development personnel should be available for consultation during integration test execution. If an error is serious enough to suspend test execution, integration and development task leads will coordinate immediate analysis. Otherwise, integration test errors will be discussed at the daily status presentation before being entered into ICAS.

3.6 Subsystem Integration Test Summary

After software turnover to system test, the integration team will prepare an informal report of testing. The report is an overall evaluation of the formal integration test phase, including a summary of problems encountered and resolved, and a list of outstanding problems.

3.7 Lessons Learned

Immediately after turnover to system test, the integration test team holds a Lessons Learned meeting. Any problems encountered during the test phase are discussed, with an eye toward identifying process changes that can be implemented for the next integration test phase. Meeting minutes are prepared, listing action items and due dates. The integration task lead is responsible for follow-up for action items and process improvement initiatives. Meeting minutes will also be stored on server for history.

3.8 Subsystem Integration Test Schedule

The following is a schedule for subsystem integration test planning and execution for the three LPS builds. The date listed for each item is the COMPLETION date, unless otherwise indicated. All dates are relative to the date of the turnover to System Test. The planned turnover dates will be maintained in the Build Implementation Plan.

ACTIVITY	COMPLETION DATE
Test case selection	turnover - 9 weeks
Test data identification	turnover - 9 weeks
Integration test walkthrough	turnover - 8 weeks
Test environment setup/cleanup	turnover - 5 weeks
Test data generation/verification	turnover - 4 weeks
First turnover of subsystem(s)	turnover - 4 weeks
Last turnover of subsystem(s)	turnover - 2 weeks
Subsystem integration	turnover - 1 week
String testing	turnover - 1 day
Turnover package preparation	turnover - 1 day
Turnover to system test	turnover + 0 days
Confidence testing	turnover + 3 days

Lessons learned meeting	turnover + 1 week
Test summary report	turnover + 2 weeks

Section 4 -- System Testing

4.1 Objectives

The primary objective of system testing is to verify the functional, operational, and performance requirements allocated to the LPS. The Landsat 7 Processing System (LPS) Functional and Performance Specification requirements are the basis for the system test design defined in this plan.

In addition to requirements verification, system testing includes running operational scenarios, simulating a "day in the life" for operational software. End-to-end testing is also performed, encompassing LPS software functions and interfaces with external elements of the Landsat 7 ground system.

4.2 Test Case Definition

The Landsat 7 Processing System (LPS) Functional and Performance Specification, the Landsat 7 Processing System (LPS) Software Requirements Specification, and the Landsat 7 Processing System (LPS) Operations Concept documents are the basis for defining system test sets and test cases. System test cases are mapped to the F&PS requirements and are designed to exercise LPS processes in a manner that verifies that allocated requirements are satisfied. For requirements that are satisfied by operational procedures rather than software, test cases follow the scenarios defined in the operations concept document.

Functional tests focus on valid, invalid or incomplete keyed and data inputs to the system. Performance tests focus on data volumes capabilities, response times and data throughput. Operational scenario testing exercises system functions in combinations that are most likely to occur during daily operations.

System test cases are grouped under functional test sets. A system test set consists of a purpose, high level description of the tests to be performed, testing dependencies, test methods and an overview of expected output. Test cases include a description of the test and data, and a list of the requirements satisfied with successful execution. Because test sets group similar tests together, the data, resources and expected results are defined at the test set level. When individual tests diverge from the test set norm, more details of resources and output are included at the test case level.

4.2.1 Test Ordering and Dependencies

System test cases are grouped into test sets by LPS function. Tests within a set are generally independent of each other, but may follow a data flow that defines the order of test execution. Likewise, test sets must be executed in a specific order (e.g., data receipt must precede data processing), unless simulations are used to satisfy the prerequisites for test execution. Details of test dependencies are documented along with the description of each test set.

4.3 Test Procedures

The system test procedures consist of step-by-step instructions for executing individual test cases. Test procedures are written based on details of LPS operations documented in the user's guide. An initial set of procedures is developed for a build, presented at a Test Procedures Walkthrough, and modified as necessary before testing begins. The set of test procedures baselined for the build is added as an appendix to the I &T Plan. During system test execution, the procedures are red-lined to adjust for software changes and omissions in test steps. At the end of the system test phase, the procedures are updated per red-lines, and form the basis of the next build's test procedures. At the completion of the last system test phase, the I & T Plan contains the comprehensive test procedures for the LPS.

Test procedures typically consist of

- test set name and number
- test case number and description
- test execution dependencies
- test data description
- test tools required
- check-off list of steps to execute test
- check-off list of expected results
- success criteria, where applicable
- requirements satisfied by successful execution

Because many LPS requirements map to low-level tasks within the LPS processing function, system test cases will not have a one-to-one correspondence to requirements. For more complicated test cases, the test steps will be mapped to expected results, which will be mapped to individual requirements. This design will allow test cases and procedures to follow the operation of the LPS system without complicating the analysis required to assess test results.

4.4 Test Procedures Walkthroughs

A Test Procedures Walkthrough is held for each build as part of the test planning phase. The TPW is conducted to ensure that the proposed test procedures adequately describe the operation of the system and verify the system requirements implemented for the current build. The walkthrough should be attended by representatives from development, integration and test, systems engineering, product assurance and project management.

A review package, containing all test procedures to be executed during the system test phase, should be distributed a week before the scheduled TPW; reviewers should have sufficient time to prepare comments and questions before the meeting.. At the walkthrough, an overview of the test methods, test data, tools and other resources employed for testing the build is presented by system test personnel. Review of the individual procedures is driven by comments on test execution steps, expected output and/or success criteria.

4.5 System Test Readiness Review

A System Test Readiness Review is conducted to assess the readiness of the system, the test environment, and the system test team for system testing. The degree of formality of the STRR depends on scope, size, and criticality of the system. While the first build of LPS is not a formal release, its functional scope and size merit a formal STRR.

The STRR should be conducted at least one week before the scheduled turnover to system test, but less mature systems (i.e., early builds) should be held as far in advance of the turnover as possible. The system test team facilitates the meeting, but the readiness of different areas will be presented by the following groups:

(1) Software support - software development
(2) Subsystem I&T results - integration test
(3) Test environment - configuration management

If the STRR assessment shows that any element reviewed is not ready, a plan for implementing corrective action should be established, within a time frame that allows for reassessment before turnover.

4.6 Test Execution

The test sets and test cases documented in this test plan are high level descriptions of test to be performed during the lifecycle of the LPS project. Details of test cases will be documented in the test procedures associated with each build.

All tests are executed as described in the test procedures. Problem reports are generated when discrepancies exist between the expected output and the actual results of the test. Expected output is noted in each test procedure, and concurred upon by the project at the Test Procedures Walkthrough.

The success criteria stated for each test case act as a pass/fail indicator for statistical reporting during the system test phase. Success criteria are also reviewed at the Test Procedures Walkthrough.

In addition to providing test execution steps and criteria for assessing test results, the test procedures are used as test logs. The procedure is written as a checklist, which allows the tester to indicate exactly where problems were encountered, which test step or output was the basis for ICCR generation, the date of test execution, the status of test results, and any comments pertinent to the test.

4.6.1 Functional and Performance Tests

Functional tests are defined to verify functional requirements specified in the LPS F&PS. Nominal input conditions are defined for every functional requirement test, abnormal input conditions (e.g., incorrect input, incomplete or invalid data) are defined for appropriate and representative tests. Functional tests are executed during all builds, and are the primary focus of the first two builds of the LPS. They also become the basis for building the regression test set for subsequent builds.

For a given build, functional tests are scheduled to exercise

- new functions
- most used functions
- most important functions
- hardest to develop functions, if applicable
- least time to develop functions, if applicable
- least unit-tested functions, if applicable.

Performance tests are defined to verify the performance requirements specified in the LPS F&PS, such as

- overall performance (system monitoring and utilization)
- data volume

- storage limits
- usability (operator interface and user's guide)
- reliability
- recovery and availability
- response times.

System tests for performance verification are executed in builds 2 and 3, when the software is fully implemented and the hardware is available. Performance tests cases are defined mainly in the System Integration Test section of this document.

4.6.2 External Interfaces

These tests are defined to verify the LPS software that supports interfaces with external elements of the Landsat 7 Ground System, as specified in the LPS F&PS. Tests will be executed in all builds of the LPS; build 1 testing is expected to utilize interface simulators, while subsequent builds will require coordination with the facilities with which the LPS will interface. However, if the external facilities are avaliable before build 2 testing, build 1 software will be used for external interface test.

4.6.3 Operational Scenarios

Operational scenario testing is defined to verify the requirements in the LPS F&PS that are not satisfied by software or hardware implementation. In addition, scenarios described in the LPS Operations Concept document are tested to verify the feasibility of the workload assigned to the LPS operators.

Operational, performance, and end-to-end tests are combined to provide "day in the life" testing in which the LPS is up-and-running for a minimum of 24 hours.

4.6.4 End-to-End Testing

End-to-end testing verifies the LPS' capability to support input-to-output testing. Data are processed through the entire system, from the receipt of LGS data, through L0R data processing and file generation, to the transfer of those files to the LP DAAC. Different scenarios are established for nominal processing, reprocessing and processing data with correctable errors. End-to-end testing is also employed in system integration testing and performance verification.

4.6.5 Confidence and Regression Tests

Both confidence and regression tests use a baselined set of data to make a quick, initial assessment of the LPS software installed in the system test environment.

For confidence testing, nominal data successfully processed in the integration test environment are passed to the system test group to process. Duplicate results in the two environments provide a high level of confidence in the build process. Confidence tests are executed starting with the first turnover of the first build.

In contrast, regression tests are executed for turnovers made after a stable software baseline has been verified in the system test environment. After successful execution, nominal functional test cases may be moved to the regression test set. The data associated with the original tests and the test results are configured for use and comparison in later regression testing. Changes to the LPS software that affect the results of baselined test cases may affect the regression test set, requiring certain tests to be suspended from regression testing until a new data set has been tested and configured.

After an initial regression set can be baselined, regression tests are executed for each subsequent turnover to the system test group. A subset of the regression test set is identified based on the contents of the individual turnover; the full regression set is executed prior to delivery for released builds. Wherever possible, regression testing is automated to run without impacting test time.

4.7 Verification Methodology

Most requirements will be verified by test, in which expected results derived from the requirements are the basis for assessing actual test results. Those requirements not verified by testing are verified by inspection, analysis or demonstration. Data formats and report contents are typically verified by inspecting the output for compliance with the applicable requirements. Inspection is also performed on hardware, code, program design language (PDL), and other documentation. Performance requirements are verified by test followed by analysis to further interpret the results. Requirements relating to the operator interface may also be verified by human factors analysis. Requirements that are satisfied by implementing operational procedures can be verified by demonstrating that the procedure are practical and possible. The verification methodology for tests is noted along with the test set description.

4.8 Analysis and Reporting

Most of the analysis performed during the system test phase is a comparison of data files. For builds that employ regression testing, the output files from newly executed tests are analyzed with respect to the configured output of the regression tests. Output data from individual tests are analyzed with respect to the input data, for both nominal and anomalous data handling. All data generated by the LPS are collected and analyzed to determine whether output files meet the established criteria documented in the system test procedures.

After completion of the system test phase for a build, each test procedure is reviewed for completeness, and the status of its associated requirement(s) is documented for the system test verification matrix. The RTM tool will be used to generate the traceability matrix, and to store the status of test execution.

Test progress statistics are collected on a daily basis, status reports are prepared on a weekly basis and included in the weekly reports to GSFC. At the conclusion of all testing for a build, a test status summary report is generated, based on the results of requirements verification during the system test phase. The test report for an undelivered build is less formal, and available within two weeks of the end of testing. Reports for software release testing are available within a month of test completion.

4.9 Anomaly Reporting and Change Control

Any tests outlined in this document that fail to generate the expected results may be rerun to verify that, if a problem does exist, it can be recreated and the test results maintain consistency. Problems that are not repeatable are still documented and noted as nonrepeatable. ICCRs are documented through ICAS and reviewed by software engineering to determine whether they are in the scope of the LPS requirements and design implementation. ICAS maintains a database of the ICCRs for approvals, analysis, implementation, tracking and report generation.

Problems found during system test are resolved by interim turnovers of modified software, depending on the severity of the problem and the test time remaining until the delivery deadline. During test phases, the Configuration Review Board (CRB) will meet as often as necessary to make timely decisions on the disposition of software problems encountered during testing. The decision whether to change the system test software with emergency or patch turnovers will be made by the CRB, however, no changes will be made to the system test baseline without a formal turnover of configured software.

4.10 Requirements Traceability

To ensure that all requirements for the LPS are addressed, each system test case is mapped, using the Requirements and Traceability Management (RTM) tool, to the requirement(s) it is designed to verify. The mapping, by test case, is part of the System Test Sets section of this document. Appendix B contains the test verification matrix generated from RTM. For each requirement, the matrix lists test case(s) needed to completely satisfy the requirement, the build(s) in which each test is executed, and the test results.

The initial test execution schedule is derived from the Landsat 7 Processing System (LPS) Build Implementation Plan, and is subject to change. Updates to the LPS System Test Verification Matrix are made after each Test Procedures Walkthrough (to update the test execution schedule), and after the completion of system testing (to update the test results). A requirements verification matrix is included in the delivery package for software releases.

4.11 Lessons Learned

At the end of build testing, or after software delivery to the customer, the system test team holds a Lessons Learned meeting. Any problems encountered, as well as successful procedures employed during the test phase are discussed. Test management personnel are in attendance and are responsible for resolution of issues and implementation of changes that require management level approval. Meeting minutes are prepared with action items and due dates, which should be scheduled for changes to be implemented before the next test execution phase.

4.12 Test Assumptions and Constraints

Specific test assumptions are covered in the listing of resources required for each test set (i.e., it is assumed the specified resources will be available before the related tests are executed). Additionally, the system test group expects

- to have access to any tools used by development and integration for testing
- that a draft of the user's guide will be available approximately 4 months before the first turnover from integration test as a resource for test procedures development
- that the software turned over to system test will include "best guess" values for the static data in the database, and tools to automatically load that data

- o that the minimum amount of system time (8 hours a day, 5 days a week) will be available during the test execution phase
- o that development and/or integration test personnel will be available to support the system test team

The system test phase of the LPS is planned based on the build implementation plan and published schedules for development and test. Test schedules are affected by changes in turnover schedules, system availability, and additional release requirements.

Because the Landsat 7 satellite will not be launched before the completion of the last test phase, the test data sets in this test plan are simulated. Though the data will come from various sources (GTSIM, Landsat 4/5 data conversion tools, instrument I&T, spacecraft I&T), none is guaranteed to duplicate data that will be received after launch. The results of testing are necessarily limited by the available data.

4.13 System Test Activities Schedule

The following is a schedule for system test planning and execution for the three LPS builds. The date listed for each item is the COMPLETION date, unless otherwise indicated. All dates are relative to the date of the turnover to System Test. The planned turnover dates will be maintained in the Build Implementation Plan.

ACTIVITY	COMPLETION DATE
Test case selection	turnover - 8 weeks
Test data identification	turnover - 8 weeks
Test procedures generation	turnover - 5 weeks
Test Procedures Walkthrough	turnover - 4 weeks
Test data generation/verification	turnover - 4 weeks
Test environment setup/cleanup	turnover - 2 weeks
System Test Readiness Review	turnover - 1 week
Dry run of the build	turnover - 1 week
1st turnover from Integration Test	turnover + 0 days

Confidence testing	turnover + 3 days
Last turnover from Integration Test	turnover + 6 weeks
LPS System Demonstration	turnover + 6 weeks
Regression testing before delivery	turnover + 8 weeks
Test execution	turnover + 8 weeks
Delivery package preparation	turnover + 8 weeks
Release to GSFC	turnover + 8 weeks
Lessons learned meeting	turnover + 9 weeks
Test summary report	turnover + 12 weeks

Section 5 -- Configuration and Data Management

This section describes the configuration activities that pertain to the testing phases of the LPS. The Configuration Management section of the Baseline Engineering and System Test Department is predominantly responsible for configuration control.

5.1 Test Tools

Tools developed for testing the LPS will be documented (e.g. description, location, release used in) and place in a configured software library for reuse and access by all LPS personnel. Changes to the configured test tools will be controlled and tracked, and CM personnel will be responsible for maintenance of the configured test tool library.

Test tools will be made available to the operations staff when the LPS software is released. The tools will be provided as engineering versions and will not be formally delivered. Any modifications made by EROS Data Center (EDC) personnel to the tools are outside the scope of SEAS test, maintenance and configuration control.

5.2 Test Data

LPS test data will be configured and associated with a release baseline, but will not be a deliverable item. CM personnel will be responsible for maintaining a test data catalog and configured test tool library, and any changes to the configured test data will be tracked.

The data catalog will contain information on data characteristics, status, and location; and will be accessible to all LPS personnel. After test data files have been validated (e.g., it has been determined that they contain the desired characteristics), they will be stored in a common location to allow easy access by multiple groups (i.e., development, integration test, and system test). Some small test data files will be kept on-line to allow quick access. Larger files will be kept on tape and will be stored in CSC's Greentec IV facility in order to maintain configuration control.

5.2 Test Environments

This section describes the configuration of the integration and system test environments. A test environment consists of hardware elements, system software and COTS products including the operating system, database, and interface software not part of the LPS application software.

5.3.1 Setup and Verification

The test groups, with the support of the System Engineering and System Administration staff, will be responsible for the identification, setup, and verification of their respective test environments.

CM will be responsible for documenting each test environment (i.e., hardware elements and software versions) via a checklist audit prior to the start of a test period. The test environment is always a configured environment and changes to it must be documented and approved via a CCR.

5.3.2 Cleanup and Maintenance

After testing has been completed, the test environment must be prepared for the beginning of the next test phase.

- o new regression data should be identified, cataloged and stored on tape
- o all data, files, logs, messages, or mail that support test assessment or problem report analysis should be written to tape or hardcopy, marked with the appropriate reference, and stored
- o all other (expendable) data files and log files should be purged from the test baseline
- o the baseline database should be archived (by ORACLE tool)
- o baseline LPS software should be backed up to tape, along with the database archive. The test group should maintain the current baseline and one previous; tapes from earlier baselines can be recycled

Maintenance of the test environment is the responsibility of the test group. Limitations on test resources may require environment cleanup to be done during the test phase. (i.e., releasing unnecessarily used disk space) to avoid impacts on test execution.

5.4 Software Turnovers

Upon successful completion of the development phase per build, the Project Configuration Review Board will determine if the software for a planned release or a software patch is ready to be promoted to the next phase of testing (i.e. integration test or system test). Once approval is granted an Electronic Software Change Notice will be prepared to transition the software to the next phase.

Upon the receipt of the Electronic Software Change Notice the LPS Configuration Management Assistant will promote the units to the appropriate test team, build the executables, install them in the test environment, and invoke the system. If each step is successful the system will be turned over to the appropriate test manager**.**

5.5 Delivery Packaging

The preparation of the delivery package is a team effort involving software development, system engineering, configuration management, integration and system test personnel. A Delivery Agreement Memorandum (DAM) approved by the ATR drives the contents (i.e., new development and CCRs) of the delivery. The turnover packages (internal deliveries) from a given release become the delivery package draft, which is maintained by the CM group. Before the final document is produced, an internal delivery package review is held to ensure the completeness and accuracy of the contents.

Section 6 -- Test Specifications

All test sets for the LPS subsystem integration and system test efforts are described in this section.

6.1 Subsystem Integration Test Sets

The following sections describe the test sets and cases planned for integrating the software subsystems of the LPS.

6.1.1 IT Test Set 1 -- Management and Control Subsystem

Integration test set 1 tests the processing paths for the Management and Control Subsystem (MACS). This test set includes invocation by the User Interface, access to the LPS database, and invocation of Level 0R subsystems. Since MACS invokes the Level 0R subsystems, integration test set 1 incorporates the invocation by MACS of the Raw Data Processing Subsystem (RDPS), the Major Frame Processing System (MFPS), the Payload Correction Data Processing Subsystem (PCDS), and the Image Data Processing Subsystem (IDPS). In addition, integration test set 1 tests the generation of metadata files and the invocation of the LPS Data Transfer Subsystem (LDTS).

The following test cases are planned for MACS integration testing.

01. To invoke RDPS, MFPS, PCDS, and IDPS subsystems without error.

02. To generate a metadata file without error and to invoke LDTS.

03 To invoke the Level 0R subsystems with errors and to terminate Level 0R processing prematurely.

04 To encounter errors during generation of the metadata files that inhibits invocation of LDTS.

05. To invoke the Level 0R subsystems without error and to stop Level 0R processing via the User Interface.

06. To request a tape copy of Raw Wideband Data that has successfully been captured.

07. To request a restage of Raw Wideband Data and to stop the restage process via the User Interface.

6.1.2 IT Test Set 2 -- Raw Data Processing Subsystem

Integration test set 2 tests the processing paths for the Raw Data Processing Subsystem (RDPS). This test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of blocks of CADUs. Since RDPS reads raw wideband data files, Integration test set 2 incorporates the testing of input/output functions such as opening and closing files and error conditions which may result during RDPS processing.

The following test cases are planned for RDPS integration testing.

01. To produce blocks of CADUs without errors. This test case generates blocks of CADUs which are then fed as input into a tool that enables the integration tester to determine if the contents of the CADUs are correct.

02. To be unable to open the Raw Wideband Data file.

03. To be unable to access the database. Default values should be employed.

04. To be unable to access shared memory. Processing should terminate immediately.

05. To identify and flag invalid frame synchronization markers during CADU generation.

06. To process blocks of raw wideband data that contain CRC, RS, and BCH errors.

07. To process blocks of raw wideband data that produce CADUs with fill data. The output from this test case is run through the integration test tool to determine if the CADUs are correct.

08. To process blocks of raw wideband data with correctable and non-correctable bit slips, frame synchronization errors, and CRC errors.

09. To be unable to write to a CADU trouble file.

6.1.3 IT Test Set 3 -- Major Frame Processing Subsystem

Integration test set 3 tests the processing paths for the Major Frame Processing Subsystem (MFPS). This integration test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of aligned band data files, calibration data files, and mirror scan correction data files. Since MFPS uses the blocks of CADUs from RDPS as input, integration test set 3 incorporates the testing of shared memory functions. Since MFPS updates quality and accounting information in the

data base, integration test set 3 incorporates the interface between MFPS and the database.

The following test cases are planned for MFPS integration testing.

01. To provide error free CADUs to MFPS and to use the output from MFPS as input to a tool which allows the integration tester to determine if the output produced by MFPS is correct.

02. To provide CADUs with scan bit flips, VCID changes, and erroneously placed end of contact markers.

03. To provide CADUs with erroneous major frame times and invalid major frame synchronization patterns.

04. To provide CADUs which contain gaps within a major frame.

05. To provide CADUs containing old data.

06. To provide CADUs with bad VCIDs and missing VCIDs.

07. To be unable to access the database for quality and accounting statistic updates and/or subinterval identifier generation.

08. To be unable to access shared memory.

6.1.4 IT Test Set 4 -- Payload Correction Data Processing Subsystem

Integration test set 4 tests the processing paths for the Payload Correction Data Processing Subsystem (PCDS). This integration test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of scene data in the form of PCD cycles and bands present information. Since PCDS uses extracted PCD data from MFPS, integration test set 4 incorporates the testing of shared memory functions. In addition, integration test set 4 incorporates the interface with the LPS database.

The following test cases are planned for PCDS integration testing.

01. To provide error free PCD cycles and bands present data, and to use the output from PCDS as input to a tool which allows the integration tester to determine if the output produced by PCDS is correct.

02. To provide PCD information words with frame synchronization bit errors.

03. To provide information to PCDS indicating that VCDU minor frames are missing.

04. To provide PCD information words with sufficient errors to result in the generation of threshold error messages.

05. To fail to write to the PCD output file causing a premature termination of the PCDS subsystem.

06. To fail an update to the LPS database.

07. To be unable to access shared memory.

6.1.5 IT Test Set 5 -- Image Data Processing Subsystem

Integration test set 5 tests the processing paths for the Image Data Processing Subsystem (IDPS). This integration test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of band files and browse files. Since IDPS uses aligned band data from MFPS and scene data from PCDS, integration test set 5 incorporates the testing of shared memory functions. In addition, integration test set 4 incorporates the interface with the LPS database.

The following test cases are planned for IDPS integration testing.

01. To provide error free aligned bands data and error free scene data to produce band files and browse files and to use the output from IDPS as input to a tool which allows the integration tester to determine if the output produced by IDPS is correct.

02. To provide aligned band data with an invalid format identifier.

03. To provide corrupted scene data causing data gaps during IDPS processing.

04. To fail a write to a browse file causing IDPS to exit prematurely.

05. To be unable to access shared memory.

6.1.6 IT Test Set 6 -- LPS Data Transfer Subsystem

Integration test set 6 tests the processing paths for the LPS Data Transfer Subsystem (LDTS). This integration test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of a Data Availability Notice (DAN). Since LDTS uses information from the LPS database, integration test set 6 incorporates the interface with the LPS database.

The following test cases are planned for LDTS integration testing.

01. To provide all data files necessary for the generation of a DAN. This DAN becomes input to a tool which allows the integration tester to determine if the DAN generated by LDTS is correct.

02. To provide files that cannot be opened or read by LDTS.

03. To produce conditions that prohibit the connection to the destination.

04. To produce conditions that inhibit successful transmission of files to the destination.

05. To produce condition that inhibit LDTS to store a DAN.

6.1.7 IT Test Set 7 -- Raw Data Capture Subsystem

Integration test set 7 tests the processing paths for the Raw Data Capture Subsystem (RDCS). This integration test set includes invocation by the Management and Control Subsystem, access to the LPS database, and generation of a data capture file. Since RDCS retrieves information from the LPS database, integration test set 7 incorporates the interface with the LPS database.

The following test cases are planned for RDCS integration testing.

01. To provide 4 strings of raw data for capture by RDCS. This raw data should be error free so that the captured data file becomes input to a tool which allows the integration tester to determine if the data capture file is correct.

02. To provide conditions that prevent the creation of a data capture file.

03. To provide conditions that prevent the database from being used.

04. To provide raw data that results in errors exceeding threshold limits.

05. To provide conditions resulting in a data base update error.

06. To provide conditions which prevent the invocation of RDCS by MACS.

6.2 System Test Sets

The following sections describe the test sets and cases planned for system testing of the LPS. Because the test cases are grouped by logical sets, test dependencies, expected output and resources required to test are defined on the test set level. When required for clarification, details are included at the test case level.

6.2.1 System Test Set 1 -- Raw Wideband Data Receipt

Purpose: To ensure that wideband data from the Landsat 7 satellite are received by the LPS from the Landsat 7 Ground Station (LGS).

Description: This test set verifies that data can be received by the LPS, by LGS output channel and by contact, including simultaneous receipt of up to 4 channels of data. Tests are included to verify that data capture starts and ends according to schedule, or immediately upon an operator directive. Performance tests verify the size and rate requirements for raw wideband data receipt.

Dependencies: There are no test set dependencies for data receipt, however, data receipt tests are expected to be run in conjunction with data storage tests (test set 2).

Expected Output:
- Automatic data receipt starts and stops according to predefined schedule
- Manual data receipt starts and stops according to operator directives
- Data are received through the LGS-LPS interface
- Receipt function does not terminate until scheduled or manual end of process
- Gaps in data flow do not affect receipt function
- A contact period of up to 14 minutes can be received by the LPS
- Data are received at 75 Mbps
- Data received by the LPS are not changed during the transfer
- Receipt of data on each LPS string is independent from any other string

Resources Required:
- Raw data capture subsystem software
- Management and control subsystem software
- LGS-LPS hardware interface
- LGS or facility simulator
- LPS front end RAID
- Landsat 7 raw wideband data files
- GTSIM and GTEDIT
- Digital Linear Tape (DLT) drive

Test Method: Functional, interface and performance tests verified by test execution and inspection.

F&PS Req	Test Case	Test Description	Additional Results
3.1.11 3.3.6. 8	1.01	Start manual data receipt per operator directive.	
3.2.1 3.3.1. 1 3.3.1. 2 4.2.1	1.02	Receive nominal raw wideband data from LGS, by output channel, for a contact period	
3.1.10 3.1.11 3.3.6. 8	1.03	Manually stop data receipt function during data capture.	
3.3.1.11	1.04	Start automatic data receipt per schedule stored in LPS database.	
3.3.6. 8	1.05	Stop automatic data receipt per schedule stored in LPS database.	
3.1. 3 3.1.11 3.2.1	1.06	Receive 4 simultaneous wideband data inputs from LGS (nominal data).	
3.3.1. 1 4.3.1	1.07	Receive 14 minutes of raw wideband data from the LGS (nominal data).	
3.1.11 3.3.6. 8	1.08	Schedule automatic data receipt to stop before end of data transmission; use manual start of receipt function to capture remaining data.	- two separate data receipt sessions create two separate raw data files on the LPS
3.3.1.12	1.09	Receive nominal data from LGS, transmission errors cause gaps in data flow (i.e., periodically receive "noise" rather than data).	- LPS receives full scheduled contact as single raw data file
3.3.1. 1 3.3.1.12 3.3.1.13	1.10	Receive nominal data from LGS, interface disconnects during transmission	- Scheduled receipt continues unless operator manually overrides

6.2.2 System Test Set 2 -- Raw Wideband Data Storage

Purpose: To ensure that wideband data received from the LGS are stored on the LPS, on both online (disk) and short term (tape) media and can be retrieved from storage.

Description: This test set verifies that data received from the LGS are captured by the LPS and stored in the front end RAID. Tests in this set verify that data stored on disk can be written to tape, and that data can be restored back from tape to disk .

Dependencies: Tests to verify data storage to disk must be run in conjunction with data receipt tests (test set 1). Other test cases in this set require some sequential ordering (e.g., write to tape before read from tape).

Expected Output:
- Data received from LGS are automatically written to the capture RAID
- Separate contact receipts create separate stored data files
- A single contact file can be created for up to 14 minutes of received data
- The front end RAID will accommodate 3 contact files (maximum input data size)
- LPS tape will accommodate at least one 14 minute contact file
- Contact files are created for data received at 75 Mbps
- Data received at the LPS are not changed during transfer
- Data copied from disk to tape, or tape to disk, are not changed
- Storage of data on each LPS string is independent from any other string

Resources Required:
- Raw data capture subsystem software
- Management and control subsystem software
- LGS-LPS hardware interface
- LGS or facility simulator
- LPS front end RAID
- LPS Digital Linear Tape(s)
- Landsat 7 raw wideband data files
- GTSIM and GTEDIT
- tool -- formatted dump of stored data
- LPS Digital Linear tape (DLT) drive

Test Method: Functional and performance tests verified by test execution and inspection.

F&PS Req	Test Case	Test Description	Additional Results
3.3.1. 3 3.3.1. 4	2.01	Write nominal raw wideband data to disk, by output channel, by contact period.	

3.1. 3 3.3.1. 4	2.02	Write 4 channels of nominal raw wideband data from LGS simultaneously to corresponding disks.	
3.3.1. 4	2.03	Write nominal raw wideband data to disk, for 14 minute contact period.	
3.3.1. 7	2.04	Write raw wideband data from disk to tape, by output channel, by contact period.	
3.1. 8 3.3.1. 5 3.3.1. 6 3.3.1. 9	2.05	Write raw wideband data from tape to disk.	
3.3.1. 4	2.06	Write raw wideband data to disk with insufficient space to accommodate complete file.	- LPS notifies operator with error messages and alarms
3.3.1. 9	2.07	Write data from tape to disk already storing that data (i.e., original file has not been deleted)	- LPS prompts operator to confirm before overwriting exiting file

6.2.3 System Test Set 3 -- Nominal Data Processing

Purpose: To ensure that Level 0R processing can be performed on raw wideband data received from the LGS.

Description: This test set verifies that raw wideband data stored on the LPS can be processed to create the expected LPS files. Because LPS processing is executed as a single function at the system level, the test cases in this set are defined by the characteristics of the data to be processed. The test procedures will map test steps to evaluation criteria and to requirements so that partial implementation of processing can be tested and evaluated.

Dependencies: Data receipt and storage test (test sets 1 and 2) must be executed before data can be processed.

Expected Output:
- CCSDC AOS Grade 3 service is performed on all data
- Data are processed by subinterval
- ETM+ reverse scans are handled correctly

- Instrument data bands are deinterleaved correctly
- Instrument files are created for each band
- MSCD files are created for each subinterval
- Calibration files are created for each subinterval
- PCD files are created for each subinterval
- Status data from the VCDU are appended to major frames
- Browse files are generated for each subinterval
- Metadata files are generated for each subinterval
- All output files are correct with respect to input data

Resources Required:
- Raw data processing software
- Major frame processing software
- Payload correction data software
- Image data processing software
- Management and control software
- Front end RAID
- SGI XL CPUs
- Back end RAID
- tools -- to produce formatted dumps of input data, L0R files, browse files, PCD files, metadata files, database tables, and contents of memory shared between subsystems
- data -- characteristics specified in each test case
- GTSIM and GTEDIT

Test Method: Functional and performance tests verified by test execution and analysis.

F&PS Req	Test Case	Test Description	Additional Results

3.1.4 3.1.8 3.3.1.5 3.3.2.1 3.3.2.2 3.3.2.7 3.3.2.11 3.3.2.14 3.3.2.15 3.3.2.16 3.3.2.17 3.3.2.18 3.3.2.20 3.3.2.21 3.3.2.22 3.3.2.23 3.3.2.24 3.3.2.25 3.3.2.28 3.3.2.29 4.3.3	3.01	Process a file of nominal raw wideband data.	
3.3.2.24	3.02	Process a file of nominal raw wideband data that is one scene in length.	
3.1.4 3.3.1.5 3.3.2.1 3.3.2.2 3.3.2.4 3.3.2.7 3.3.2.11 3.3.2.14 3.3.2.15 3.3.2.16 3.3.2.17 3.3.2.18 3.3.2.20 3.3.2.21 3.3.2.22 3.3.2.23 3.3.2.24 3.3.2.25 3.3.2.28 4.3.3	3.03	Process a 14 minute file of nominal raw wideband data.	
3.3.2.24	3.05	Process raw wideband data that crosses a day boundary.	
3.3.2.24	3.06	Process raw wideband data that crosses a month boundary.	

3.3.2.24	3.07	Process raw wideband data that crosses a year boundary.	
3.3.2.24	3.08	Process raw wideband data that crosses the boundary from February 28 to February 29.	
3.3.2.24	3.09	Process raw wideband data that crosses the boundary from day 365 to day 366.	
3.3.2.24	3.11	Process raw wideband data that crosses the boundary from year 1999 to year 2000.	
3.3.2.24	3.12	Process raw wideband data in which the VCDU counter rolls over.	
3.3.2.16 3.3.2.17 3.3.2.23 3.3.2.24	3.13	Process raw wideband data in which the format changes from Format 1 to Format 2.	
3.3.2.16 3.3.2.17 3.3.2.23 3.3.2.24	3.14	Process raw wideband data in which the format changes from Format 2 to Format 1.	
3.3.2.16 3.3.2.17 3.3.2.23 3.3.2.24	3.15	Process raw wideband data in which the format changes from Format 1 to Format 2 then back to Format 1.	
3.3.2.19	3.16	Process a file of nominal raw wideband data that is smaller than one scene in length.	

6.2.4 System Test Set 4 -- Anomalous Data Processing

Purpose: To ensure that the LPS processing function can handle raw wideband data containing anomalies or errors.

Description: This test set verifies the LPS can process raw wideband data with gaps, time jumps, format changes; that bit errors within the

required tolerance can be corrected; and that unrecoverable errors in data are identified and handled appropriately.

Dependencies: Data receipt and storage tests (test sets 1 and 2) must be executed before data can be processed. Nominal data (test set 3) should be processed before error data.

Expected Output:
- Both normal and inverted polarity data are handled correctly
- BCH error detection & correction is performed on both mission data and data pointer zones of the VCDU
- plus or minus 3 bit slips per CADU are corrected
- All CADUs which fail CCSDS are stored
- All CADUs which fail BCH are stored
- Fill VCDUs are deleted
- Full and partial major frames are filled
- All output files are correct with respect to input data

Resources Required:
- Raw data processing software
- Major frame processing software
- Payload correction data software
- Image data processing software
- Management and control software
- Front end RAID
- SGI XL CPUs
- Back end RAID
- tools -- to produce formatted dumps of input data, L0R files, browse files, PCD files, metadata files, database tables, and the contents of memory shared between subsystems
- data -- characteristics specified in each test case
- GTSIM and GTEDIT

Test Method: Functional and performance tests verified by test execution and analysis.

F&PS Req	Test Case	Test Description	Additional Results
3.3.2.24	4.01	Process raw wideband data that contains data in fields not used by the LPS (i.e., priority bit, replay bit, spare bits).	
3.3.2. 2 3.3.2.19 3.3.2.24	4.02	Process raw wideband data containing a partial CADU at the beginning of the data.	

3.3.2.19 3.3.2.24	4.03	Process raw wideband data containing a partial CADU at the end of the file.	
3.3.2. 2 3.3.2.24	4.04	Process raw wideband data containing a partial scan at the beginning of the file.	
3.3.2.19 3.3.2.24	4.05	Process raw wideband data containing a partial scan at the end of the file.	
3.3.2.19	4.06	Process raw wideband data containing small gaps in the data (missing minor frames).	
3.3.2.19	4.07	Process raw wideband data containing a VCDU counter rollover that occurs during a data gap.	
3.3.2.19	4.08	Process raw wideband data containing multiple VCDU counter rollovers during a data gap.	
3.3.2.19	4.09	Process raw wideband data containing large gaps in the data (missing major frames).	
3.3.2. 4 3.3.2. 6 3.3.2. 9.1	4.10	Process raw wideband data containing fewer than 4 errors in some of the data pointer fields.	
3.3.2. 4 3.3.2. 6 3.3.2. 8 3.3.2. 9.1	4.11	Process raw wideband data containing more than 3 errors in some of the data pointer fields.	
3.3.2. 6 3.3.2. 9	4.12	Process raw wideband data containing fewer than 4 bit errors in some of the minor frame data.	
3.3.2. 6 3.3.2. 8 3.3.2. 9 3.3.2.10	4.13	Process raw wideband data containing more than 3 bit errors in some of the minor frame data.	

3.3.2. 3 3.3.2. 5	4.14	Process raw wideband data containing more than one CADU of inverted polarity data.	
3.3.2. 3 3.3.2. 5 3.3.2. 8	4.15	Process raw wideband data containing less than one CADU of inverted polarity data.	
3.3.2. 1 3.3.2.24	4.16	Process raw wideband data containing forward time jumps (switch from playback to realtime data).	
3.3.2. 1 3.3.2.24	4.17	Process raw wideband data containing backward time jumps.	
3.3.2.24	4.18	Process raw wideband data containing a forward correction in the spacecraft clock.	
3.3.2.24	4.19	Process raw wideband data containing a backward correction in the spacecraft clock.	
3.3.2.15	4.20	Process raw wideband data containing bit errors in the Line Sync Code.	
3.3.2.24	4.21	Process raw wideband data containing bit errors in the time code.	
3.3.2.24	4.22	Process raw wideband data containing errors in the minor frame counter.	
3.3.2. 1	4.23	Process raw wideband data containing errors in the VCDU version number.	
3.3.2.24	4.24	Process raw wideband data containing errors in the spacecraft identifier.	
3.3.2. 1	4.25	Process raw wideband data containing errors in the VCID (values of 0, 3, 4, and 6).	

3.3.2. 1	4.26	Process raw wideband data containing errors in the VCDU Header Error Control Field.	
3.3.2. 2 3.3.2. 4	4.27	Process raw wideband data containing errors in the CADU sync pattern.	
3.3.2. 1	4.28	Process raw Format 1 wideband data containing some VCID errors (looks like Format 2).	
3.3.2. 9 3.3.2.10 3.3.2.15	4.29	Process raw wideband data containing CADU sync patterns in the Mission Data.	
3.3.2. 9 3.3.2.15	4.30	Process raw wideband data containing line sync patterns in the Mission Data.	
3.3.2. 2 3.3.2.24	4.31	Process raw wideband data containing garbled data at the beginning of the file.	
3.3.2.15	4.32	Process raw wideband data containing bit errors in the End of Line Pattern Code.	
3.3.2. 2 3.3.2.24	4.33	Process raw wideband data containing garbled data at the beginning of the file.	

6.2.5 System Test Set 5 -- Payload Correction Data Processing

Purpose: To ensure that the LPS processing includes correct handling of Payload Correction Data (PCD) bytes.

Description This test set verifies that during processing, the LPS software correctly identifies PCD bytes, assembles minor and major frames, and generates PCD files.

Dependencies: Data receipt and storage tests (test sets 1 and 2) must be executed before data can be processed.

Expected Output:
- PCD files are generated by subinterval

- Missing PCD data are filled
- All output files are correct with respect to input data

Resources Required:
- Raw data processing software
- Major frame processing software
- Payload correction data software
- Image data processing software
- Management and control software
- Front end RAID
- SGI XL CPUs
- Back end RAID
- tools -- to produce formatted dumps of input data, PCD files, database tables, and the contents of memory shared between subsystems
- data -- characteristics specified in each test case
- GTSIM and GTEDIT

Test Method: Functional and performance tests verified by test execution and analysis.

F&PS Req	Test Case	Test Description	Additional Results
3.3.3.3| 3.3.4. 1 3.3.4. 4	5.01	Process nominal raw wideband data with no PCD byte errors.	
3.3.4. 1	5.02	Process raw wideband data containing bit errors in some of the PCD sync words.	
3.3.4. 1	5.03	Process raw wideband data containing bit errors in some of the PCD minor frame sync words.	
3.3.4. 3	5.04	Process raw wideband data containing bit errors in some of the PCD minor frame counter values.	
3.3.4. 3	5.05	Process raw wideband data containing bit errors in some of the PCD major frame counter values.	
3.3.4. 1	5.06	Process raw wideband data with bit errors in the PCD time code data.	

3.3.2.25 3.3.2.29 3.3.4. 3	5.07	Process raw wideband data with bit errors in the PCD calibration door activity status information.	
3.3.4. 3	5.08	Process raw wideband data with bit errors in the PCD ephemeris information.	
3.3.4. 2	5.09	Process raw wideband data containing small gaps in the PCD data (several bytes).	
3.3.4. 2	5.10	Process raw wideband data containing small gaps in the PCD data (several minor frames).	
3.3.4. 2	5.11	Process raw wideband data containing large gaps in the PCD data (several major frames).	
3.3.4. 2	5.12	Process raw wideband data with bytes missing from the PCD time code data.	
3.3.4. 2	5.13	Process raw wideband data with no PCD bytes.	

6.2.6 System Test Set 6 -- Image Data Assessment

Purpose: To ensure that browse data, cloud cover assessment, and the moving window display provide accurate views of the Landsat data.

Description: This test set verifies that the algorithms for determining cloud cover in a WRS scene, and for providing browse files and displays of the data, accurately represent the data processed by the LPS.

Dependencies: Data receipt and storage tests (test sets 1 and 2) must be executed before data can be processed.

Expected Output:
- Browse data are generated by subinterval
- Multiband browse is generated from the predetermined bands
- Browse data are generated using predefined reduction factor
- Scenes are identified correctly with respect to WRS standards

- Cloud cover assessments are accurate with respect to input data
- Cloud cover assessment is provided for both quadrants and full scenes
- Cloud cover assessment is performed using predefined comparison values
- **(moving window display)**
- All output data are correct with respect to input data

Resources Required:
- Raw data processing software
- Major frame processing software
- Payload correction data software
- Image data processing software
- Management and control software
- Front end RAID
- SGI XL CPUs
- Back end RAID
- tools -- to produce formatted dumps of input data, L0R files, database tables, and the contents of memory shared between subsystems
- data -- characteristics specified in each test case
- GTSIM and GTEDIT

Test Method: Functional tests verified by analysis and inspection.

F&PS Req	Test Case	Test Description	Additional Results
3.3.3.1 3.3.3.3 3.3.3.4 3.3.3.5	6.01	Generate a browse file from nominal raw wideband data processing.	
3.3.3.1 3.3.3.3 3.3.3.4 3.3.3.5	6.02	Generate a browse file from processing data containing small gaps (missing minor frames).	
3.3.3.1 3.3.3.3 3.3.3.4 3.3.3.5	6.03	Generate a browse file from processing data with large gaps (missing several major frames).	
3.3.3.1 3.3.3.3 3.3.3.4 3.3.3.5	6.04	Generate a browse file from data containing large gaps (missing full scene or more).	
3.3.4. 8 3.3.4. 9 3.3.4.10	6.05	Process data containing a WRS scene of consistent characteristics (i.e., ocean, desert)	

3.3.4. 8| 3.3.4. 9 3.3.4.10	6.06	Process data containing a WRS scene consisting only of clouds.	
3.3.4. 8 3.3.4. 9 3.3.4.10	6.07	Process data containing a WRS scene with approximately 50% cloud cover, confined to two quadrants.	
3.3.4. 8 3.3.4. 9 3.3.4.10	6.08	Process data containing a WRS scene with approximately 50% cloud cover, spread across all four quadrants.	
3.3.4. 8 3.3.4. 9 3.3.4.10	6.09	Process data containing a WRS scene with approximately 50% cloud cover, confined to two quadrants.	

6.2.7 System Test Set 7 -- System Monitoring and Control

Purpose: To ensure that the LPS software provides the capability to monitor operations and provides operator controls.

Description: This test set verifies that the LPS provides displays and messages to indicate progress and status of the system and software; and that the LPS operator can start up the software, change certain configuration parameters, override automated functions, and display statistical information

Dependencies: Monitoring tests are run in conjunction with all other test sets (functions to be monitored). Test cases related to data transfer and reporting require the data to be available (test sets 3-7). Manual override tests have the same dependencies as the sets that test the nominal function.

Expected Output:
- Information messages are provided during all LPS functions
- Error messages are provided as appropriate for all LPS functions
- Statistical updates for receipt and processing functions are provided
- Operator can override data receipt start/stop
- Operator can override data processing start/stop
- Operator can override data transfer
- Operator can override data deletion

Resources Required:
- Management and control software

- Operator terminal and printer
- Operator interface display
- resources required from test sets executed simultaneously

Test Method: Functional and performance tests verified by test execution, analysis and inspection.

F&PS Req	Test Case	Test Description	Additional Results
3.1.12	7.01	Monitor LPS system initialization.	
3.1.10.1 3.1.11	7.02	Startup the LPS interface for each operational string.	
3.1.12	7.03	Monitor automatic data receipt (nominal data) from startup to termination.	
3.1.11 3.3.6. 8 3.3.6. 9	7.04	Manually start data processing of nominal data.	
3.1.12	7.05	Monitor automatic data receipt (gaps in data flow during transmission) from startup to termination.	
3.1.11 3.3.6. 8	7.06	Manually terminate current data processing.	
3.1.12	7.07	Monitor automatic data receipt during which hardware interface fails.	
3.1.12	7.08	Monitor raw data written to tape.	
3.1.12	7.09	Monitor attempt to write data to unusable tape.	
3.1.12	7.10	Monitor nominal wideband data processing.	
3.1.10.3 3.1.12 3.3.6 7	7.11	Monitor processing of data with unrecoverable BCH errors.	
3.1.12 3.3.6. 7	7.12	Monitor processing of data with fill data in all PCD bytes.	
3.1.12	7.13	Monitor DAN transfer to LP DAAC.	

3.1.12	7.14	Monitor DAN transfer to LP DAAC during hardware interface failure.	
3.1.12	7.15	Monitor receipt of nominal DTA and subsequent data file deletion.	
3.1.11 3.3.5.6 3.3.5.8	7.16	Override data deletion (retain data).	
3.1.11 3.3.5.5 3.3.6. 8	7.17	Manually delete files.	
3.1.12	7.18	Monitor receipt of DTA in unreadable format.	
3.1.12	7.19	Monitor attempt to delete files with invalid names (incorrect DTA or already-deleted files)	
3.3.6. 1	7.20	Enter configuration information in LPS setup tables.	
3.3.6. 1	7.21	Modify existing configuration in LPS setup tables.	
3.3.6. 1	7.22	Enter processing thresholds in LPS setup tables.	
3.3.6. 1 3.3.6. 6	7.23	Modify existing processing thresholds in LPS setup tables.	
3.3.6. 1	7.24	Enter error thresholds in LPS setup tables.	
3.3.6. 1 3.3.6. 6	7.25	Modify existing error thresholds in LPS setup tables.	
3.3.6. 4	7.26	Display return link quality and accounting data.	
3.3.6. 4	7.27	Display level 0R quality and accounting data.	
3.3.6. 4	7.28	Display PCD quality and accounting data from metadata.	
3.3.6. 4.1	7.29	Print return link quality and accounting data.	
3.3.6. 4.1	7.30	Print level 0R quality and accounting data.	

3.1.10 3.1.10.3 3.1.12	7.31	Monitor LPS functions from receipt through DAN transfer.	
3.1.10.3 3.3.6. 1 3.3.6. 8 3.3.6. 9	7.32	Enter invalid data in LPS setup tables (character input to value fields)	
3.1.10.3 3.3.6. 1	7.33	Enter invalid data in LPS setup tables (values out of defined range).	
3.1.10.2 3.1.11	7.34	Shut down the LPS interface for each operational string.	

6.2.8 System Test Set 8 -- Diagnostics and Test

Purpose: To ensure that the LPS software provides the capability to test, diagnose, and maintain the operational system.

Description: This test set verifies that the LPS provides a capability to execute diagnostics and test its functions without impacting normal operations.

Dependencies: Diagnostics and test cases have the same dependencies as the functions that are being tested (refer to applicable test sets)

Expected Output:
- Each LPS function can be tested without impacting operations
- Functional tests duplicate operational use of each function
- Diagnostics tests check and report on all system operations

Resources Required:
- Management and control software
- Operator interface display
- resources required from related test sets

Test Method: Functional tests verified by test execution, analysis and inspection.

F&PS Req	Test Case	Test Description	Additional Results

3.1.10.6 3.1.10.8 3.1.19	8.01	Test data receipt/LGS-LPS interface hardware.	
3.1.10.6 3.1.10.8 3.1.19	8.02	Test DAN transfer/LPS-LP DAAC hardware interface.	
3.1.10.6 3.1.10.8 3.1.19	8.03	Test data receipt/file creation on disk	
3.1.10.6 3.1.10.8 3.1.19	8.04	Test tape drive	
3.1.10.6 3.1.10.8 3.1.19	8.05	Test data processing startup and termination.	
3.1.10.6 3.1.10.8 3.1.19	8.06	Test DAN generation.	
3.1.10.6 3.1.10.8 3.1.19	8.07	Test DTA receipt.	
3.1.10.4 3.1.10.7	8.08	Execute system diagnostics.	

6.2.9 System Test Set 9 -- Reporting

Purpose: To ensure that the LPS generates summary and accounting reports as required.

Description: This test set verifies that the LPS generates data receipt and transfer summaries, and quality and accounting statistics on the correct basis (e.g., contact or subinterval), with accurate statistics.

Dependencies: Data receipt, data processing, and data transfer tests (sets 1, 3, 10 min) must be executed before the associated report data are available.

Expected Output:
- Report data are accumulated on the required basis
- Reports contain all data required
- Report data are stored for a minimum of 30 days

- Operator can request report generation for display
- Operator can request report generation for hard copy
- Report data are accurate with respect to Landsat data received/processed/transferred

Resources Required:
- Management and control software
- Operator interface display
- resources required from related test sets

Test Method: Functional tests verified by analysis and inspection.

F&PS Req	Test Case	Test Description	Additional Results
3.3.1.10 3.3.1.10.1	9.01	Generate data receipt summary - nominal data receipt.	- Summary can be generated within 5 minutes of end of data receipt -Summary is generated on contact period basis
3.3.4. 5 3.3.4.14	9.02	Generate PCD quality and accounting data from nominal data processing.	- Data are accumulated on a scene and on a subinterval basis
3.3.4. 5 3.3.4.6 3.3.4.15	9.03	Generate processed PCD quality and accounting data from nominal data processing.	- Data are accumulated on a scene and on a subinterval basis
3.1. 6 3.3.2.13 3.3.6. 2	9.04	Generate return link quality and accounting data from nominal data processing.	- Data are accumulated on a contact period basis.
3.1. 7 3.3.2.26 3.3.6. 3	9.05	Generate level 0R quality and accounting data from nominal data processing.	- Data are accumulated on a subinterval basis.
3.3.4.11 3.3.4.12 3.3.4.13 3.3.4.14 3.3.4.15 3.3.4.16	9.06	Generate metadata information from nominal data processed.	

3.3.5.7	9.08	Generate transfer summary report from nominal transfer.	- Data are accumulated on the basis of transfers per day.
3.1. 7	9.09	Generate data receipt summary -- receipt stopped manually during capture.	
3.1. 6 3.3.2.13	9.10	Generate return link quality and accounting data from data processed with uncorrectable BCH errors.	
3.1. 7 3.3.2.26	9.11	Generate level 0R quality and accounting data from data processed with missing major frames.	
3.1. 7 3.3.2.26	9.12	Generate level 0R quality and accounting data from processing data with bit errors in minor frame counters.	
3.3.4. 5 3.3.4.14 3.3.4.15	9.13	Generate PCD quality and accounting data from processing data containing all fill for PCD bytes.	
3.3.4. 5 3.3.4.14 3.3.4.15	9.14	Generate PCD quality and accounting data from processing data with bit errors in PCD major frame counters.	

6.2.10 System Test Set 10 -- Data Transfers

Purpose: To ensure that the LP DAAC can retrieve files from the LPS.

Description: This test set verifies that the LPS generates a Data Availability Notice (DAN) accurately listing the files available from a contact period, that the DAN is transferred to the LP DAAC, that the LP DAAC can access the indicated files for transfer, that the LPS can receive and process the Data Transfer Acknowledgment (DTA) sent from the LP DAAC. This test

set also verifies the management of the LPS files with regard to deletion after transfer.

Dependencies: Data files must be available from data processing (test sets 3-6) in order to execute data transfer tests.

Expected Output:
- DAN is generated and sent to LP DAAC immediately upon file generation
- DTAs can be received from LP DAAC
- DTAs can be processed by the LPS
- Files successfully transferred to the LP DAAC are deleted from the LPS
- File transfers are made on a contact basis
- File deletion is performed on a contact basis
-The DAN-DTA process can be repeated to resolve incomplete transfers
- Transfer problems are resolved by operator intervention rather than endless DAN-DTA transfers

Resources Required:
- Management and control software
- LPS data transfer software
- LPS-LP DAAC hardware interface
- LP DAAC or facility simulator
- LPS back end RAID
- LPS output files
- Operator interface display
- resources required from related test sets

Test Method: Functional, interface and performance tests verified by execution.

F&PS Req	Test Case	Test Description	Additional Results
3.1. 3 3.2.2 3.3.5.1	10.01	Verify DAN is automatically generated and sent to the LP DAAC when data processing is completed.	
3.2.2 3.3.5.3	10.02	Verify DTA is successfully received from the LP DAAC, and applicable files are deleted from the LPS.	
3.1. 3 3.2.2 4.2.2	10.03	Verify LP DAAC can successfully retrieve LPS files.	
3.1.11 3.3.6. 8 3.3.6. 9	10.04	Disable transfer of DAN.	

3.1.11 3.3.6. 8 3.3.6. 9	10.05	Enable previously disabled DAN.	
3.2.2	10.06	Manually delete LPS files after DAN is successfully transferred, before files are retrieved by the LP DAAC.	
3.2.2	10.07	Manually delete LPS files before DTA is received from LP DAAC.	

6.3 System Integration Test Sets

System Integration Tests are intended to verify the LPS as an operational system. Interfaces with the LGS and the LP DAAC are intended to be run with the hardware and software of the actual facilities. Data should originate at the LGS (though it will be simulated test data), process through the LPS, and be retrieved by the LP DAAC.

Tests are expected to run on the operational configuration of hardware (4 main strings and 1 backup string), or as close an approximation as is available.

System Integration Tests are used specifically to verify performance requirements, though data limitations may not allow verification of LPS processing the maximum daily load.

6.3.1 System Integration Test Set 1 -- Operational Scenarios and Procedures

F&PS Req	Test Case	Test Description	Additional Results
3.1.23	S1.01	Set up backup string for training while LPS is operational.	
3.1.20	S1.02	Install modified LPS software on backup string for test in operational setting.	
3.1.10	S2.01	Reconfigure LGS-LPS interface to backup string.	
3.1.10.5 3.1.14 3.2.21	S2.02	Substitute backup string for one current string.	
3.1.22	S2.03	Perform maintenance on backup string, substitute for operational string, repeat for all strings.	

3.2.4	S2.04	Receive processing parameters from the IAS via FAX or other hardcopy.	
3.2.3 3.3.1.10. 1	S2.05	Provide data receipt information to the MOC via voice or FAX.	
3.3.1. 8	S2.06	Store tapes (30 days) recorded with raw wideband data.	
4.3.6	S2.07	Store raw wideband data on tape for 30 days.	
3.3.5.2	S2.08	DAN transfer fails, contact LP DAAC operator.	
3.3.5.2 3.3.5.4 4.1. 7	S2.09	DTA not received within 8 hours of transfer, contact LP DAAC operator.	
3.3.1.11	SI1.01	LGS failure recovery test	
3.1.3 3.1.21	SI1.02	LPS failure recovery test	
3.1.8 3.3.1.9	SI1.03	LP DAAC failure recovery test	

6.3.2 System Integration Test Set 2 -- Performance

F&PS Req	Test Case	Test Description	Additional Results
4.1. 3	S3.01	Receive 100 GB data from LGS in one day.	
4.1. 3	S3.02	Process 100 GB data from LGS in one day.	
4.1. 6	S3.03	Process data at 12 Mbps (average aggregate rate).	
4.3.2	S3.04	Store raw wideband data for three contact periods (maximum size input data)	
4.1. 4	S3.05	Write daily volume of received data (100 GB) to tape within 16 hours of receipt (16 hrs after end of last receipt).	
4.1. 4	S3.06	Process daily volume of received data (100 GB) within 16 hours of receipt (16 hrs after end of last receipt).	
4.1. 5	S3.07	Reprocess up to GB of raw data (in addition to 100 GB daily volume processing) in one day.	
3.1. 1 3.1. 2 3.1 5	S3.08	Receive and process data for 6 contacts a day, 24 hours a day for 7 days	

Section 7 -- Test Resources Requirement

7.1 Test Environment

The following subsections describe the hardware and software configurations from which testing will be performed.

7.1.1 Hardware configuration

The system administrator will set up and configure all LPS hardware elements, including system software, for the test environment.

The operational LPS will consist of five independent hardware strings. The Build Implementation Plan defines the hardware and network configuration requirements for each build. In general, each test group will utilize an independent hardware string. For some tests, multiple hardware strings will be required and before the first release, tests will be conducted utilizing the four operational hardware strings to verify that the planned configuration can be operated as planned.

Many tests will be executed via remote Xterminals at CSC's Greentec IV facility. Some tests will require access to hardware located at GSFC (e.g. tape drives) and consequently will be executed at GSFC. Each tester will require simultaneous access to a separate Xterminal.

7.1.2 Software configuration

Separate software environments will be established for each of the test groups. Executable software for each build will be located in separate directory structures and the individual test groups will be able to concurrently execute tests using the software in these directories. Each test group will also have a separate database instance. The development organization will provide a set of "suggested" initial values for all static database tables. These values will be changed by the test groups as necessary to conduct the tests. After the initial static values have been loaded, the contents of the database instances will be maintained by the individual test groups.

7.2 Test Tools

Test tools will be used wherever possible to automate and support the test effort. The following subsections describe details of the test tools that are planned.

7.2.1 Drivers and Stubs

A driver will be required to simulate transfer of data from the LGS.

An LP DAAC simulator will be required to verify the Data Availability Notice (DAN)/Data Transfer Acknowledgment (DTA) exchange process. This simulator will receive DANs, validate DANs, and return DTAs. The simulator will have the capability to either automatically send a valid DTA or to send a DTA with predefined errors.

Because Build 1 will not produce output files that can be analyzed, stubs will be required to dump data from shared memory for analysis of some of the subsystems' output.

During integration testing, stubs and drivers will be required to simulate subsystems that have not yet been developed.

7.2.2 Data Dump and Analysis Tools

The following data dump tools will be developed to assist in the analysis of test output:

Tool	Description
Sync dump	The sync dump tool will produce both frame-synchronized and non-synchronized hexidecimal dumps of up to 10 CADUs of input data.
Input data summary dump	The input data summary dump tool will produce a formatted, frame synchronized, dump of input CADUs such that a summary of each CADU is placed on each line of output. Summary information to be included consists of the sync pattern, the VCDU header fields, the PCD status words, and the data pointer.

Band file dump	The band file dump tool will allow the user to choose whether a summary dump or a full dump is desired for a range of scan lines. If a summary dump is chosen, the tool will produce an ASCII formatted dump such that each line of the tool's output contains all of the band file information from a single scan except for the scan data line (pixel data). The full dump option will follow each line of the summary dump with a hexidecimal dump of the image data.
Calibration file dump	The calibration file dump tool will allow the user to choose whether a summary dump or a full dump is desired for a range of scan lines. If a summary dump is chosen, the tool will produce an ASCII formatted dump such that each line of the tool's output contains all of the calibration file information from a single scan except for the calibration data line (pixel data). The full dump option will follow each line of the summary dump with a hexidecimal dump of the image data.
MSCD file dump	The MSCD file dump tool will produce an ASCII formatted dump for a user-specified range of scan lines such that each line of the tool's output contains all of the MSCD file information from a single scan.
PCD file dump	The PCD file dump tool will produce an ASCII formatted dump for a user-specified range of PCD major frames. The user will be able to specify whether or not the PCD major frame data is to be included in the output.
Browse file dump	The browse file dump tool will generate a formatted ASCII dump of the text information associated with the browse file and a hexidecimal dump of the image data associated with the file.
Database table dump	The database dump tool will present the user with a list of the LPS database tables. The user will then be able to select which table to dump and whether all fields or a predetermined subset of the fields are desired. Alternatively, the user may execute a script that will dump the contents of each table into a file.
RDPS shared memory dump	The MACS subsystem will pass to the RDPS shared memory dump tool the information that it needs to attach to the correct area of shared memory. The shared memory dump tool will then attach, open, and read the memory blocks containing RDPS output and will write a formatted dump of this data to a file. For each CADU output by RDPS, the dump program will extract the value for each field and write the data to a file. A subset of the minor frame data for each CADU will be written to the file.

MFPS shared memory dump	The MACS subsystem will pass to the MFPS shared memory dump tool the information that it needs to attach to the correct area of shared memory. The shared memory dump tool will then attach, open, and read the memory blocks containing MFPS output and will write a formatted dump of this data to a file.

Metadata files will be in an ASCII PVL format that can be viewed by using the UNIX cat or more commands. No dump tool is needed.

The gtedit utility will be used to view hexidecimal dumps of trouble files.

A comparator (e.g., UNIX diff program) will be used for regression tests to compare the output from the current build to that of previous builds.

7.2.3 Test Execution Scripts

UNIX shell scripts will be developed to:

reinitialize the test environment (e.g., set database values back to a known starting point)
combine frequently executed commands
automate regression tests

7.2.4 Test Tool Design and Reuse

To the greatest extent possible, attempts will be made to reuse tools and scripts that already exist. Other MOSDD projects (e.g., Pacor II, DDF, DESIM) will be investigated to determine if any tools or scripts exist that can be reused. Portions of the DESIM tools will be reused for the LP DAAC simulator. Additionally, all LPS tools and scripts that are either acquired or developed for use on the LPS will be placed in a common area so that they may be shared among all of the development and the test groups.

7.2.5 Test Tool Development, Verification and Configuration

Test tools will be developed by:

GTSIM:	GTSIM personnel
LGS simulator	NASA personnel
LP DAAC simulator	SEAS development/SEAS system test
dump/analysis tools	SEAS development/SEAS system test

All test tools must be verified and placed under configuration control before testing begins for the build in which the tools will be used. This configuration control may be formal (i.e., the tools are placed into the project's configuration control system) or informal (i.e., changes to the tools are controlled by limiting write access to the executable files). Typically, tools that have been developed to dump data files or simulate external systems will be formally controlled, while test execution scripts will be informally controlled.

7.3 Test Data

The following subsections describe details of the data that are planned for testing the LPS.

7.3.1 Test Data Determination

Test data will be designed with several purposes in mind:

- to verify that the LPS correctly processes data containing the characteristics that are nominally received from the LGS
- to verify the functionality of the LPS software
- to stress the LPS data processing algorithms
- to verify that the LPS correctly processes data containing the errors that are most likely to occur during the mission's lifetime

Some test data of the nominal size will be required. Most will be smaller in order to allow for quicker generation, evaluation, and ease of storage.

7.3.2 Test Data Generation and Verification

The Generic Telemetry Simulator (GTSIM) (and its associated utility programs) will be used to generate simulated data in the format that will be received from the LGS. The current GTSIM hardware cannot support files larger than 2 GB. As test data files greater than 7 GB will be required for LPS verification, GTSIM will be ported to one of the LPS computers in order to perform generation of these large files. Before being used for LPS verification, each test data file will be examined to verify that it contains the expected test data characteristics.

Some error conditions will be introduced into the test data as it is generated by GTSIM, additional errors will be introduced after the data has been generated. The GTEDIT tool will be used to introduce individual bit errors into particular data records.

Any data that are received from prelaunch tests with the Landsat spacecraft, or interface tests with the LGS, will be saved and re-used for future testing of the LPS. Data from previous Landsat missions will be extracted from archive, attitude and ephemeris data will be converted to the Landsat 7 format, and the reformatted data will be inserted into the payload correction portion of the test data produced by GTSIM.

7.4 Personnel Resources

The following subsections describe details of the staff planned for testing the LPS and otherwise supporting the test teams.

7.4.1 Test Personnel

At the end of each development phase, a team will be assembled to perform the subsystem integration and testing. This team will be formed from the development, system engineering, and system test organizations.

System tests will typically be conducted by a team of three people. These people will perform the build/release test planning, test data and tool preparation, and test execution.

All test personnel will require skills in UNIX and Oracle. Training will be required for personnel who do not already have the necessary experience.

7.4.2 Support Personnel

In order to write effective test procedures, test personnel will have to work closely with development personnel during the test planning phase for each build. The developers will ensure that the testers fully understand how to operate the system and analyze test results before testing begins.

During each test execution phase, it is expected that a primary contact person will be designated from each of the following areas to support test problem resolution and facility scheduling:

software development
system engineering
configuration management
product assurance
system administration
LGS
LP DAAC

7.5 Schedule Requirements

For each build, as module testing is completed for each subsystem, the subsystems will be promoted to the integration test environment. As each subsystem is added, tests will be conducted to integrate the new subsystem with the rest of the system. After all subsystems have been promoted to this level, a nominal 2 week integration test period will begin.

The system test execution phase will be conducted over a 2 month period, following successful completion of integration testing. The first few days are typically used to perform and verify the installation of software in the system test environment. Confidence test are executed to verify that the build was successful, and that the integration test and system test environments are configured correctly. At least one week before each official software turnover to System Test, an "engineering" version of the software will be made available to the system test team so that dry runs of the test procedures can be performed.

ACRONYMS

AOS	Advanced Orbiting Systems
ATR	Assistant Technical Representative
BCH	Bose-Chaudhuri-Hocquenghem (error detection and correction scheme)
CADU	Channel Access Data Unit
CCR	Configuration Change Request
CCSDS	Consultative Committee for Space Data Systems
CM	Configuration Management
CRB	Configuration Review Board
CRC	Cyclic Redundancy Check
CSC	Computer Sciences Corporation
DAN	Data Availability Notice
DFCB	Data Format Control Book
DTA	Data Transfer Acknowledgment
ECS	EOSDIS Core System
EDC	EROS Data Center
EOSDIS	Earth Observation Data Information System
EROS	Earth Resources Observation System
ETM+	Enhanced Thematic Mapper Plus (instrument)
F&PS	Functional and Performance Specification
GSFC	Goddard Space Flight Center
ICAS	Interactive CCR Automation System
ICCR	Internal Configuration Change Request
ICD	Interface Control Document
IDD	Interface Description Document
LGS	Landsat 7 Ground Station
LPS	Landsat 7 Data Processing System
LP DAAC	Land Processes Distributed Active Archive Center
Mbps	megabits per second
MITG	Mission Integration Technology Group
MOC	Mission Operations Center
MO&DSD	Mission Operations and Data Systems Directorate
MOU	Memorandum of Understanding
MSCD	Mirror Scan Correction Data
NASA	National Aeronautics and Space Administration

PCD	Payload Correction Data
PDL	Program Design Language
PVCS	Polytron Version Control System
RCS	Revision Control System
RS	Reed-Solomon (error detection and correction scheme)
RTM	Requirements & Traceability Management
SEAS	Systems, Engineering, and Analysis Support
STRR	System Test Readiness Review
SSDM	SEAS System Development Methodology
TPW	Test Procedures Walkthrough
VCDU	Virtual Channel Data Unit
WRS	Worldwide Reference System

LPS Requirements Traceability Matrix

Build columns indicate tests scheduled to be executed in each build (indicated by an "X" in the column), tests deferred to the next build (indicated by a "D"), and whether the test results pass (indicated by a "P") or fail (indicated by an "F") the success criteria established in the test procedures.

F&PS Req	Test Case	Build 1	Build 2	Build 3
3.1. 1	S3.08			
3.1. 2	S3.08			
3.1. 3	1.06			
	2.02			
	10.01			
	10.03			
	SI1.02			
3.1. 4	3.01			
	3.03			
3.1. 5	S3.08			
3.1. 6	9.04			
	9.10			
3.1. 7	9.05			
	9.09			
	9.11			
	9.12			
3.1. 8	2.05			
	3.01			
	SI1.03			
3.1.10	1.03			
	7.31			
	S2.01			
3.1.10.1	7.02			
3.1.10.2	7.34			
3.1.10.3	7.11			
	7.31			
	7.32			
	7.33			
3.1.10.4	8.08			
3.1.10.5	S2.02			
3.1.10.6	8.01			
	8.02			

	8.03			
	8.04			
	8.05			
	8.06			
	8.07			
3.1.10.7	8.08			
3.1.10.8	8.01			
	8.02			
	8.03			
	8.04			
	8.05			
	8.06			
	8.07			
3.1.11	1.01			
	1.03			
	1.06			
	1.08			
	7.02			
	7.04			
	7.06			
	7.16			
	7.17			
	7.34			
	10.04			
	10.05			
3.1.12	7.01			
	7.03			
	7.05			
	7.07			
	7.08			
	7.09			
	7.10			
	7.11			
	7.12			
	7.13			
	7.14			
	7.15			
	7.18			
	7.19			
	7.31			

3.1.14	S2.02			
3.1.19	8.01			
	8.02			
	8.03			
	8.04			
	8.05			
	8.06			
	8.07			
3.1.20	S1.02			
3.1.21	S2.02			
	SI1.02			
3.1.22	S2.03			
3.1.23	S1.01			
3.2.1	1.02			
	1.06			
3.2.2	10.01			
	10.02			
	10.03			
	10.06			
	10.07			
3.2.3	S2.05			
3.2.4	S2.04			
3.3.1. 1	1.02			
	1.07			
	1.10			
3.3.1. 2	1.02			
3.3.1. 3	2.01			
3.3.1. 4	2.01			
	2.02			
	2.03			
	2.06			
3.3.1. 5	2.05			
	3.01			
	3.03			
3.3.1. 6	2.05			
3.3.1. 7	2.04			
3.3.1. 8	S2.06			
3.3.1. 9	2.05			
	2.07			
	SI1.03			

3.3.1.10	9.01			
3.3.1.10.1	9.01			
	S2.05			
3.3.1.11	1.04			
	SI1.01			
3.3.1.12	1.09			
	1.10			
3.3.1.13	1.10			
3.3.2. 1	3.01			
	3.03			
	4.16			
	4.17			
	4.23			
	4.25			
	4.26			
	4.28			
3.3.2. 2	3.01			
	3.03			
	4.02			
	4.04			
	4.27			
	4.31			
	4.33			
3.3.2. 3	4.14			
	4.15			
3.3.2. 4	3.01			
	3.03			
	4.10			
	4.11			
	4.27			
3.3.2. 5	4.14			
	4.15			
3.3.2. 6	4.10			
	4.11			
	4.12			
	4.13			
3.3.2. 7	3.01			
	3.03			
3.3.2. 8	4.11			
	4.13			

	4.15			
3.3.2. 9	4.12			
	4.13			
	4.29			
	4.30			
3.3.2. 9.1	4.10			
	4.11			
3.3.2.10	4.13			
	4.29			
3.3.2.11	3.01			
	3.03			
3.3.2.12				
3.3.2.13	9.04			
	9.10			
3.3.2.14	3.01			
	3.03			
3.3.2.15	3.01			
	3.03			
	4.20			
	4.29			
	4.30			
	4.32			
3.3.2.16	3.01			
	3.03			
	3.13			
	3.14			
	3.15			
3.3.2.17	3.01			
	3.03			
	3.13			
	3.14			
	3.15			
3.3.2.18	3.01			
	3.03			
3.3.2.19	3.16			
	4.02			
	4.03			
	4.05			
	4.06			
	4.07			

	4.08			
	4.09			
3.3.2.20	3.01			
	3.03			
3.3.2.21	3.01			
	3.03			
3.3.2.22	3.01			
	3.03			
3.3.2.23	3.01			
	3.03			
	3.13			
	3.14			
	3.15			
3.3.2.24	3.01			
	3.02			
	3.03			
	3.05			
	3.06			
	3.07			
	3.08			
	3.09			
	3.11			
	3.12			
	3.13			
	3.14			
	3.15			
	4.01			
	4.02			
	4.03			
	4.04			
	4.05			
	4.16			
	4.17			
	4.18			
	4.19			
	4.21			
	4.22			
	4.24			
	4.31			
	4.33			

3.3.2.25	3.01			
	3.03			
	5.07			
3.3.2.26	9.05			
	9.11			
	9.12			
3.3.2.28	3.01			
	3.03			
3.3.2.29	3.01			
	5.07			
3.3.3.1	6.01			
	6.02			
	6.03			
	6.04			
3.3.3.3	5.01			
	6.01			
	6.02			
	6.03			
	6.04			
3.3.3.4	6.01			
	6.02			
	6.03			
	6.04			
3.3.3.5	6.01			
	6.02			
	6.03			
	6.04			
3.3.4. 1	5.01			
	5.02			
	5.03			
	5.06			
3.3.4. 2	5.09			
	5.10			
	5.11			
	5.12			
	5.13			
3.3.4. 3	5.04			
	5.05			
	5.07			
	5.08			

3.3.4. 4	5.01			
3.3.4. 5				
3.3.4. 5	9.02			
	9.03			
	9.13			
	9.14			
3.3.4. 6	9.03			
3.3.4. 7				
3.3.4. 8	6.05			
	6.06			
	6.07			
	6.08			
	6.09			
3.3.4. 9	6.05			
	6.06			
	6.07			
	6.08			
	6.09			
3.3.4.10	6.05			
	6.06			
	6.07			
	6.08			
	6.09			
3.3.4.11	9.06			
3.3.4.12	9.06			
3.3.4.13	9.06			
3.3.4.14	9.02			
	9.06			
	9.13			
	9.14			
3.3.4.15	9.03			
	9.06			
	9.13			
	9.14			
3.3.4.16	9.06			
3.3.5.1	10.01			
3.3.5.2	S2.08			
	S2.09			
3.3.5.3	10.02			
3.3.5.4	S2.09			

3.3.5.5	7.17			
3.3.5.6	7.16			
3.3.5.7	9.08			
3.3.6. 1	7.20			
	7.21			
	7.22			
	7.23			
	7.24			
	7.25			
	7.32			
	7.33			
3.3.6. 2	9.04			
3.3.6. 3	9.05			
3.3.6. 4	7.26			
	7.27			
	7.28			
3.3.6. 4.1	7.29			
	7.30			
3.3.6. 5				
3.3.6. 5.1				
3.3.6. 6	7.23			
	7.25			
3.3.6. 7	7.11			
	7.12			
3.3.6. 8	1.01			
	1.03			
	1.05			
	1.08			
	7.04			
	7.06			
	7.16			
	7.17			
	7.32			
	10.04			
	10.05			
3.3.6. 9	7.04			
	7.32			
	10.04			
	10.05			
3.3.6.10				

4.1. 3	S3.01			
	S3.02			
4.1. 4	S3.05			
	S3.06			
4.1. 5	S3.07			
4.1. 6	S3.03			
4.1. 7	S2.09			
4.1. 8				
4.1. 9				
4.1.10				
4.1.11				
4.1.12				
4.1.13				
4.2.1	1.02			
4.2.2	10.03			
4.3.1	1.07			
4.3.2	S3.04			
4.3.3	3.01			
	3.03			
4.3.4				
4.3.5				
4.3.6	S2.07			
4.4.1				
4.4.2				
4.4.3				